

GrdPI_Update

Функция(метод) **GrdPI_Update** позволяет перезаписывать данные защищенной ячейки или определитель алгоритма. При этом определитель алгоритма можно перезаписать только целиком, а в защищенной ячейке можно перезаписывать любой участок данных.

C

```
int GRD_API GrdPI_Update(
    HANDLE hGrd,
    DWORD dwItemNum,
    DWORD dwAddr,
    DWORD dwLng,
    void *pData,
    DWORD dwUpdatePsw,
    DWORD dwMethod,
    void *pReserved
);
```

<i>hGrd</i>	хэндл, через который будет выполнено обновление				
<i>dwItemNum</i>	номер защищенной ячейки/алгоритма для которой будет производиться обновление				
<i>dwAddr</i>	смещение в определителе защищенной ячейки/алгоритма, начиная с которого будет производиться обновление				
<i>dwLng</i>	длина блока данных, которые будут обновлены				
<i>pData</i>	указатель на буфер, содержащий данные для обновления				
<i>dwUpdatePsw</i>	пароль для обновления. Если пароль не используется, параметр должен быть равен 0				
<i>dwMethod</i>	метод обновления данных. Задается одной из констант GrdUM_XXX <table><tr><td>GrdUM_MOV</td><td>Данные из буфера <i>pData</i> заменяют старые данные</td></tr><tr><td>GrdUM_XOR</td><td>Данные из буфера <i>pData</i> складываются со старыми данными по модулю 2</td></tr></table>	GrdUM_MOV	Данные из буфера <i>pData</i> заменяют старые данные	GrdUM_XOR	Данные из буфера <i>pData</i> складываются со старыми данными по модулю 2
GrdUM_MOV	Данные из буфера <i>pData</i> заменяют старые данные				
GrdUM_XOR	Данные из буфера <i>pData</i> складываются со старыми данными по модулю 2				
<i>pReserved</i>	зарезервировано. Параметр должен быть равен NULL				

Возможные ошибки

GrdE_AccessDenied	неверный пароль обновления, доступ к ячейке запрещен
GrdE_StatusUnchangeable	достигнут лимит количества ошибок, статус ячейки изменить нельзя. Доступ к ячейке на чтение заблокирован
GrdE_NoService	для данного типа ячейки не предусмотрена операция обновления
GrdE_InactiveItem	ячейка в деактивированном состоянии, обновление выполнить невозможно
GrdE_Overbound	адрес для обновления выходит за пределы определителя
	Набор ошибок Guardant API

GrdPI_Update позволяет менять определитель аппаратного алгоритма или данные защищенной ячейки (**rs_KI**) после указания пароля обновления. При этом определитель аппаратного алгоритма можно менять только целиком. При попытке изменить часть определителя возвращается код ошибки [GrdE_InvalidArg](#). В защищенной ячейке можно менять любой участок данных.

Обновление данных в дескрипторе ячейки можно производить в том случае, когда она находится в активном состоянии. Для обновления можно потребовать указать пароль *dwUpdatePsw*, который задается при создании ячейки или алгоритма. При создании ячейки в GRDUTIL пароль требуется по умолчанию.

Если алгоритм или ячейка находятся в активном состоянии (Active) и был задан правильный пароль на обновление, функция возвращает [GrdE_Ok](#) и производится обновление данных, начиная со смещения в определителе ячейки *dwAddr*. При этом данные из буфера *pData* длиной *dwLng* в зависимости от метода обновления *dwMethod* либо замещают старые данные, либо складываются со старыми данными по модулю 2. Максимальное значение *dwLng* - 255 байт.

Функции [GrdPI_Read](#), **GrdPI_Update** в случае ключей **Guardant Sign/Time** не имеют ранее существовавшего ограничения в 255 байт на длину считываемого/перезаписываемого определителя алгоритма/защищенной ячейки. Это связано с отсутствием в указанных ключах ограничения на размер определителя.

Если смещение *dwAddr* задано слишком большим, возвращается ошибка [GrdE_Overbound](#).

Если сумма *dwAddr+dwLng* выходит за доступные для чтения пределы, данных записывается ровно столько, сколько можно записать. При этом функция возвращает [GrdE_OK](#) и никаких других кодов ошибки не генерируется.

Если при попытке обновления активной ячейки был указан неверный пароль *dwUpdatePsw*, функция **GrdPI_Update** возвращает ошибку [GrdE_AccessDenied](#). При этом автоматически в ключе уменьшается счетчик ошибок доступа к ячейке, который задается при создании ячейки. Когда значение счетчика ошибок доходит до 0, статус ячейки становится неизменяемым, то есть ячейку больше невозможно активировать /деактивировать/читать/менять ее значение. Однако если алгоритм находился в состоянии **Active**, функция [GrdTransform](#) будет нормально работать.

C#

```
public static GrdE GrdPI_Update(Handle grdHandle, GrdAlgNum algNum, uint addr, byte[] data, uint updatePsw, GrdUM method)
private static unsafe GrdE GrdPI_Update(IntPtr hAddress, int algNum, uint addr, byte[] data, uint updatePsw, int method)
```

grdHandle [in]

Тип: [Handle](#)

Нэндл, с помощью которого будет выполнено обновление.

algNum [in]

Тип: [GrdAlgNum](#)

Номер аппаратного алгоритма или защищенной ячейки, для которых будет происходить обновление.

addr [in]

Тип: uint

Смещение в определителе защищенной ячейки или алгоритма, начиная с которого будет происходить обновление.

data [in]

тип: byte []

Указатель на буфер, который содержит данные для обновления.

method [in]

типы: int, [GrdUM](#)

Метод обновления данных. Задается одной из констант [GrdUM](#).

updatePsw [in]

Тип: uint

Пароль для обновления.

Возможные ошибки

GrdE.AccessDenied	неверный пароль обновления, доступ к ячейке запрещен
GrdE.StatusUnchangeable	достигнут лимит количества ошибок, статус ячейки изменить нельзя. Доступ к ячейке на чтение заблокирован
GrdE.NoService	для данного типа ячейки не предусмотрена операция обновления
GrdE.InactiveItem	ячейка в деактивированном состоянии, обновление выполнить невозможно
GrdE.Overbound	адрес для обновления выходит за пределы определителя
	Набор ошибок Guardant API

GrdPI_Update позволяет менять определитель аппаратного алгоритма или данные защищенной ячейки (**rs_K[]**) после указания пароля обновления. При этом определитель аппаратного алгоритма можно менять только целиком. При попытке изменить часть определителя возвращается код ошибки [GrdE.InvalidArg](#). В защищенной ячейке можно менять любой участок данных.

Обновление данных в дескрипторе ячейки можно производить в том случае, когда она находится в активном состоянии. Для обновления можно потребовать указать пароль *updatePsw*, который задается при создании ячейки или алгоритма. При создании ячейки в GRDUTIL пароль требуется по умолчанию.

Если алгоритм или ячейка находятся в активном состоянии (Active) и был задан правильный пароль на обновление, метод возвращает [GrdE.OK](#) и производится обновление данных, начиная со смещения в определителе ячейки *addr*. При этом данные из буфера *data* в зависимости от метода обновления (*method*) либо замещают старые данные, либо складываются со старыми данными по модулю 2. Максимальное можно обновить 255 байт данных защищенной ячейки.

Методы [GrdPI_Read](#), [GrdPI_Update](#) в случае ключей **Guardant Sign/Time** не имеют ранее существовавшего ограничения в 255 байт на длину считываемого/перезаписываемого определителя алгоритма/защищенной ячейки. Это связано с отсутствием в указанных ключах ограничения на размер определителя.

Если смещение *addr* задано слишком большим, возвращается ошибка [GrdE.Overbound](#).

Если количество записываемых данных превышает возможности записи, данных записывается ровно столько, сколько можно записать. При этом метод возвращает [GrdE.OK](#) и никаких других кодов ошибки не генерируется.

Если при попытке обновления активной ячейки был указан неверный пароль *updatePsw*, метод [GrdPI_Update](#) возвращает ошибку [GrdE.AccessDenied](#). При этом автоматически в ключе уменьшается счетчик ошибок доступа к ячейке, который задается при создании ячейки. Когда значение счетчика ошибок доходит до 0, статус ячейки становится неизменяемым, то есть ячейку больше невозможно активировать /деактивировать/читать/менять ее значение. Однако если алгоритм находился в состоянии Active, метод [GrdTransform](#) будет нормально работать.

Java

```
public static GrdE GrdPI_Update(Handle grdHandle, int itemNum, int addr, byte[] data, int updatePsw, GrdUM method)
```

grdHandle [in]

Тип: [Handle](#)

Нэндл, с помощью которого будет выполнено обновление.

itemNum [in]

Тип: int

Номер аппаратного алгоритма или защищенной ячейки, для которых будет происходить обновление.

addr [in]

Тип: int

Смещение в определителе защищенной ячейки или алгоритма, начиная с которого будет происходить обновление.

data [in]

тип: byte []

Указатель на буфер, который содержит данные для обновления.

method [in]

тип: [GrdUM](#)

Метод обновления данных. Задается одной из констант [GrdUM](#).

updatePsw [in]

Тип: int

Пароль для обновления.

Возможные ошибки

GrdE.AccessDenied	неверный пароль обновления, доступ к ячейке запрещен
GrdE.StatusUnchangeable	достигнут лимит количества ошибок, статус ячейки изменить нельзя. Доступ к ячейке на чтение заблокирован
GrdE.NoService	для данного типа ячейки не предусмотрена операция обновления

GrdE.InactiveItem	ячейка в деактивированном состоянии, обновление выполнить невозможно
GrdE.Overbound	адрес для обновления выходит за пределы определителя
	Набор функций Guardant API

GrdPI_Update позволяет менять определитель аппаратного алгоритма или данные защищенной ячейки (**rs_K[]**) после указания пароля обновления. При этом определитель аппаратного алгоритма можно менять только целиком. При попытке изменить часть определителя возвращается код ошибки [GrdE.InvalidArg](#). В защищенной ячейке можно менять любой участок данных.

Обновление данных в дескрипторе ячейки можно производить в том случае, когда она находится в активном состоянии. Для обновления можно потребовать указать пароль *updatePsw*, который задается при создании ячейки или алгоритма. При создании ячейки в **GRDUTIL** пароль требуется по умолчанию.

Если алгоритм или ячейка находятся в активном состоянии (Active) и был задан правильный пароль на обновление, метод возвращает [GrdE.OK](#) и производится обновление данных, начиная со смещения в определителе ячейки *addr*. При этом данные из буфера *data* в зависимости от метода обновления (*method*) либо замещают старые данные, либо складываются со старыми данными по модулю 2. Максимальное можно обновить 255 байт данных защищенной ячейки.

Методы [GrdPI_Read](#), **GrdPI_Update** в случае ключей **Guardant Sign/Time** не имеют ранее существовавшего ограничения в 255 байт на длину считываемого/перезаписываемого определителя алгоритма/защищенной ячейки. Это связано с отсутствием в указанных ключах ограничения на размер определителя.

Если смещение *addr* задано слишком большим, возвращается ошибка [GrdE.Overbound](#).

Если количество записываемых данных превышает возможности записи, данных записывается ровно столько, сколько можно записать. При этом метод возвращает [GrdE.OK](#) и никаких других кодов ошибки не генерируется.

Если при попытке обновления активной ячейки был указан неверный пароль *updatePsw*, функция **GrdPI_Update** возвращает ошибку [GrdE.AccessDenied](#). При этом автоматически в ключе уменьшается счетчик ошибок доступа к ячейке, который задается при создании ячейки. Когда значение счетчика ошибок доходит до 0, статус ячейки становится неизменяемым, то есть ячейку больше невозможно активировать /деактивировать/читать/менять ее значение. Однако если алгоритм находился в состоянии Active, метод [GrdTransform](#) будет нормально работать.